L 12

```
   SSSSSSSSSSSS  YYY         YYY     SSSSSSSSSSSS
   SSSSSSSSSSSS  YYY         YYY     SSSSSSSSSSSS
   SSSSSSSSSSSS  YYY         YYY     SSSSSSSSSSSS
SSS              YYY         YYY  SSS
SSS              YYY         YYY  SSS
SSS              YYY         YYY  SSS
SSS                 YYY   YYY     SSS
SSS                 YYY   YYY     SSS
SSS                 YYY   YYY     SSS
   SSSSSSSSS           YYY           SSSSSSSSS
   SSSSSSSSS           YYY           SSSSSSSSS
   SSSSSSSSS           YYY           SSSSSSSSS
            SSS        YYY                    SSS
            SSS        YYY                    SSS
            SSS        YYY                    SSS
            SSS        YYY                    SSS
            SSS        YYY                    SSS
            SSS        YYY                    SSS
SSSSSSSSSSSS           YYY        SSSSSSSSSSSS
SSSSSSSSSSSS           YYY        SSSSSSSSSSSS
SSSSSSSSSSSS           YYY        SSSSSSSSSSSS
```

```
**FILE**ID**SYSRUNDWN







  SSSSSSSS  YY      YY    SSSSSSSS  RRRRRRRR    UU      UU  NN      NN  DDDDDDDD    WW      WW  NN      NN
  SSSSSSSS  YY      YY    SSSSSSSS  RRRRRRRR    UU      UU  NN      NN  DDDDDDDD    WW      WW  NN      NN
SS            YY      YY  SS          RR      RR  UU      UU  NN      NN  DD      DD  WW      WW  NN      NN
SS            YY      YY  SS          RR      RR  UU      UU  NN      NN  DD      DD  WW      WW  NN      NN
SS              YY  YY    SS          RR      RR  UU      UU  NNNN    NN  DD      DD  WW      WW  NNNN    NN
SS              YY  YY    SS          RR      RR  UU      UU  NNNN    NN  DD      DD  WW      WW  NNNN    NN
  SSSSSS          YY        SSSSSS    RRRRRRRR    UU      UU  NN  NN  NN  DD      DD  WW      WW  NN  NN  NN
  SSSSSS          YY        SSSSSS    RRRRRRRR    UU      UU  NN  NN  NN  DD      DD  WW      WW  NN  NN  NN
        SS      YY              SS  RR  RR        UU      UU  NN    NNNN  DD      DD  WW  WW  WW  NN    NNNN
        SS      YY              SS  RR  RR        UU      UU  NN    NNNN  DD      DD  WW  WW  WW  NN    NNNN
        SS      YY              SS  RR    RR      UU      UU  NN      NN  DD      DD  WWWW  WWWW  NN      NN    ....
        SS      YY              SS  RR    RR      UU      UU  NN      NN  DD      DD  WWWW  WWWW  NN      NN    ....
SSSSSSSS        YY        SSSSSSSS    RR      RR  UUUUUUUUUU  NN      NN  DDDDDDDD    WW      WW  NN      NN    ....
SSSSSSSS        YY        SSSSSSSS    RR      RR  UUUUUUUUUU  NN      NN  DDDDDDDD    WW      WW  NN      NN    ....


LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

Table of contents

```
 (1)        38          HISTORY                  ; DETAILED
 (2)       116          DECLARATIONS
 (3)       154          EXE$RUNDWN - IMAGE RUNDOWN SYSTEM SERVICE
 (4)       527          EXE$RESETVEC - REST PROVILEGED LIBRARY VECTORS
```

```
0000              .TITLE  SYSRUNDWN IMAGE RUNDOWN SYSTEM SERVICE
0000              .IDENT  'V04-000'
0000
0000      ;
0000      ;****************************************************************************
0000      ;*                                                                          *
0000      ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0000      ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
0000      ;*  ALL RIGHTS RESERVED.                                                    *
0000      ;*                                                                          *
0000      ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
0000      ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000      ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
0000      ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
0000      ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
0000      ;*  TRANSFERRED.                                                            *
0000      ;*                                                                          *
0000      ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
0000      ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0000      ;*  CORPORATION.                                                            *
0000      ;*                                                                          *
0000      ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
0000      ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0000      ;*                                                                          *
0000      ;*                                                                          *
0000      ;****************************************************************************
0000
0000      ;++
0000      ; FACILITY: EXECUTIVE, SYSTEM SERVICES
0000      ;
0000      ; ABSTRACT:
0000      ;
0000      ; ENVIRONMENT:
0000      ;
0000      ;--
0000      ;
0000      ;
0000              .PAGE
0000              .SBTTL  HISTORY                     ; DETAILED
0000      ;
0000      ; AUTHOR:       R. HUSTVEDT       CREATION DATE: 10-OCT-76
0000      ;
0000      ; MODIFIED BY:
0000      ;
0000      ;       V03-018 WMC0002         Wayne Cardoza           28-Aug-1984
0000      ;               Don't try to delete logical names if no pointer to them.
0000      ;
0000      ;       V03-017 RAS0316         Ron Schaefer            25-Jun-1984
0000      ;               Change TMK0001 so that we delete ALL process-private
0000      ;               logical names, not just those in the process table.
0000      ;               Use new LNM internal routine LNM$DELETE_HASH, which
0000      ;               assumes that all protection/privileges have been checked.
0000      ;               This is ECO 4 in the FT2 update.
0000      ;
0000      ;       V03-016 LJK0272         Lawrence J. Kenah       10-Apr-1984
0000      ;               Reinitialize array of starting points for privileged vectors.
0000      ;
0000      ;       V03-015 TMK0001         Todd M. Katz            03-Apr-1984
```

```
0000   ;                  Change the deletion of image logical names within the process
0000   ;                  logical name table so that the $DELLNM system service is used
0000   ;                  instead of the $DELLOG system service.
0000   ;
0000   ;          V03-014 SSA0018         Stan Amway                 9-Mar-1984
0000   ;                  Rundown page fault monitoring activity.
0000   ;
0000   ;          V03-013 RSH0092         R. Scott Hanna            31-Jan-1984
0000   ;                  Modify the security auditing process rundown support to
0000   ;                  use the new IDT dedicated pages.
0000   ;
0000   ;          V03-012 CDS0001         Christian D. Saether      16-Dec-1983
0000   ;                  Add comments reflecting new use by F11BXQP of the
0000   ;                  CCB$B_AMOD field.
0000   ;
0000   ;          V03-011 RSH0067         R. Scott Hanna            22-Sep-1983
0000   ;                  Move the RDI table deallocation to earlier in the rundown.
0000   ;
0000   ;          V03-010 RSH0022         R. Scott Hanna            23-May-1983
0000   ;                  Add process rundown support to deallocate the Security
0000   ;                  Auditing Impure Data Table (IDT) and deassign the the
0000   ;                  Audit Journal.
0000   ;
0000   ;          V03-009 SRB0084         Steve Beckhardt           29-Apr-1983
0000   ;                  Added LCK$M_INVVALBLK flag to $DEQ to invalidate value
0000   ;                  blocks of locks being dequeued at high lock modes.
0000   ;
0000   ;          V03-008 JLV0231         Jake VanNoy               24-FEB-1983
0000   ;                  Add checking of CCB$V_IMGTMP flag in running down channels.
0000   ;
0000   ;          V03-007 RSH0003         R. Scott Hanna            10-Feb-1983
0000   ;                  Add rundown processing to deallocate the RDI table
0000   ;                  and zero CTL$GL_RDIPTR.
0000   ;
0000   ;          V03-006 JWH0135         Jeffrey W. Horn           23-Nov-1982
0000   ;                  Add rundown processing for CTL$GQ_P0ALLOC and
0000   ;                  CTL$GL_PRCALLCNT.
0000   ;
0000   ;          V03-005 ACG0301         Andrew C. Goldstein,      21-Oct-1982  13:39
0000   ;                  Restore FILCNTNONZ bugcheck
0000   ;
0000   ;          V03-004 WMC0001         Wayne Cardoza             26-Aug-1982
0000   ;                  Make the privileged library vector reset a subroutine.
0000   ;
0000   ;          V03-003 LJK0164         Lawrence J. Kenah         20-Apr-1982
0000   ;                  Clear byte following RSB in privileged vector rundown loop
0000   ;                  so that next activation of privileged image succeeds.
0000   ;
0000   ;          V03-002 LJK0158         Lawrence J. Kenah         9-Apr-1982
0000   ;                  Reset privileged vectors with process-specific array
0000   ;                  to allow some vectors to survive image exit.
0000   ;
0000   ;          V03-001 ACG0278         Andrew C. Goldstein,      1-Apr-1982  10:30
0000   ;                  Enable resource wait mode during rundown,
0000   ;                  disable channel deassignment bug trap
0000   ;
0000   ;
```

```
                                  0000   116          .SBTTL  DECLARATIONS
                                  0000   117
                                  0000   118  ;
                                  0000   119  ; INCLUDE FILES:
                                  0000   120  ;
                                  0000   121
                                  0000   122          $ACBDEF                              ; DEFINE AST CONTROL BLOCK
                                  0000   123          $ACMDEF                              ; DEFINE ACCOUNTING MESSAGE OFFSETS
                                  0000   124          $CCBDEF                              ; DEFINE CHANNEL CONTROL BLOCK
                                  0000   125          $IPLDEF                              ; DEFINE INTERRUPT PRIORITIES
                                  0000   126          $LCKDEF                              ; DEFINE $ENQ/$DEQ FLAGS
                                  0000   127          $NSAIDTDEF                           ; DEFINE THE SECURITY AUDITING IDT OFFSETS
                                  0000   128          $OPDEF                               ; DEFINE OPCODE SYMBOLIC CONSTANTS
                                  0000   129          $PCBDEF                              ; DEFINE PCB OFFSETS
                                  0000   130          $PHDDEF                              ; DEFINE PROCESS HEADER OFFSETS
                                  0000   131          $PRDEF                               ; DEFINE PROCESSOR REGISTERS
                                  0000   132          $RDIDEF                              ; DEFINE RIGHTS DATABASE IDENTIFIER BLOCK OF
                                  0000   133          $SSDEF                               ; DEFINE STATUS CODES
                                  0000   134
                                  0000   135  ;
                                  0000   136  ; EQUATED SYMBOLS:
                                  0000   137  ;
                        00000004  0000   138  ACMODE=4                                     ; DISPLACEMENT TO ACCESS MODE ARGUMENT
                                  0000   139
                        00000000  0000   140  NXTKVEC=0                            ;OFFSET TO NEXT FREE KERNEL VECTOR
                        00000100  0000   141  NXTEVEC=256                          ;OFFSET TO NEXT FREE EXEC VECTOR
                        00000200  0000   142  NXTRVEC=512                          ;OFFSET TO NEXT FREE RUNDWN VECTOR
                        00000300  0000   143  NXTMVEC=768                          ;OFFSET TO NEXT MESSAGE VECTOR
                                  0000   144
                                  0000   145  ;
                                  0000   146  ; STATIC DATA DEFINITIONS:
                                  0000   147  ;
                              00000000   148          .PSECT  YEXEPAGED
                                  0000   149
                                  0000   150  PROC_TABLE:                                  ; PROCESS LOGICAL NAME TABLE NAME
52 50 24 4D 4E 4C 00000008'010E0000'  0000   151          .ASCID  /LNM$PROCESS/
               53 53 45 43 4F  000E
```

```
                                   0013   154           .SBTTL  EXE$RUNDWN - IMAGE RUNDOWN SYSTEM SERVICE
                                   0013   155
                                   0013   156  ;++
                                   0013   157  ; FUNCTIONAL DESCRIPTION:
                                   0013   158  ;        EXE$RUNDWN TERMINATES ALL PENDING ACTIVITY FOR THE SPECIFIED
                                   0013   159  ;        AND ALL LESS PRIVILEGED ACCESS MODES.  ALL RESOURCES ASSOCIATED
                                   0013   160  ;        WITH THE PENDING ACTIVITY WILL BE RELEASED IF APPROPRIATE
                                   0013   161  ;
                                   0013   162  ; CALLING SEQUENCE:
                                   0013   163  ;        CALLG   ARGLIST,EXE$RUNDWN
                                   0013   164  ;
                                   0013   165  ; INPUT PARAMETERS:
                                   0013   166  ;        ACMODE(AP) - ACCESS MODE BOUNDARY FOR WHICH RUNDOWN IS TO BE PERFORMED
                                   0013   167  ;                     (MAXIMIZED WITH THE ACCESS MODE OF THE CALLER)
                                   0013   168  ;
                                   0013   169  ; IMPLICIT INPUTS:
                                   0013   170  ;        R4 - POINTER TO CURRENT PCB
                                   0013   171  ;
                                   0013   172  ; OUTPUT PARAMETERS:
                                   0013   173  ;        NONE
                                   0013   174  ;
                                   0013   175  ; IMPLICIT OUTPUTS:
                                   0013   176  ;        ALL ACTIVITY INITIATED BUT NOT COMPLETED FOR ALL ACCESS MODES
                                   0013   177  ;        HAVING THE SAME OR LESSER PRIVILEGE THAN THAT SPECIFIED WILL
                                   0013   178  ;        BE TERMINATED.
                                   0013   179  ;
                                   0013   180  ; COMPLETION CODES:
                                   0013   181  ;        SS$_NORMAL  -  SUCCESSFUL COMPLETION
                                   0013   182  ;
                                   0013   183  ; SIDE EFFECTS:
                                   0013   184  ;        NONE
                                   0013   185  ;
                                   0013   186  ;--
                                   0013   187
                             00FC  0013   188           .ENTRY  EXE$RUNDWN,^M<R2,R3,R4,R5,R6,R7>
 50   04 AC    02    00        EF  0015   189           EXTZV   #0,#2,ACMODE(AP),R0       ; FETCH ACCESS MODE ARGUMENT
                   FFE2'       30  001B   190           BSBW    EXE$MAXACMODE             ; MAXIMIZE WITH PREVIOUS
                   57    50    D0  001E   191           MOVL    R0,R7                     ; SAVE IN R7 FOR REFERENCE
          09 24 A4    16       E5  0021   192           BBCC    #PCB$V_PWRAST,PCB$L_STS(R4),10$ ; SKIP IF NO POWER FAIL AST
            00000000'9F        D4  0026   193           CLRL    @#CTL$GL_POWERAST         ; ZAP AST ADDRESS
                   38 A4       B6  002C   194           INCW    PCB$W_ASTCNT(R4)          ; AND RESTORE AST QUOTA
                                   002F   195  10$:                                       ;
                                   002F   196  ;---------------------------------------------------------------
                                   002F   197  ;
                                   002F   198  ;        Enable resource wait mode to ensure that rundown will succeed.
                                   002F   199  ;
                                   002F   200  ;---------------------------------------------------------------
                                   002F   201
                                   002F   202           $SETRWM_S WATFLG=#0               ; ENABLE WAIT MODE
                         50    DD  0038   203           PUSHL   R0                        ; SAVE RETURN STATUS
                                   003A   204
                                   003A   205  ;---------------------------------------------------------------
                                   003A   206  ;
                                   003A   207  ;        INVOKE THE USER RUNDOWN SERVICE(S)
                                   003A   208  ;
                                   003A   209  ;        The user service is invoked in KERNEL mode with a JSB and must
                                   003A   210  ;        exit with a RSB.
```

```
                                003A        ;
                                003A        ;       R4 - Pointer to current PCB
                                003A        ;       R7 - Access mode parameter to SYSRUNDWN maximized with previous mode
                                003A        ;       ACMODE(AP) - Access mode parameter to SYSRUNDWN
                                003A        ;
                                003A        ;---------------------------------------------------------------
                                003A
       55   00000000'9F    D0   003A                MOVL    @#CTL$GL_USRUNDWN, R5   ; GET PER-PROCESS USER RUNDOWN VECTOR
                     02    13   0041                BEQL    20$                     ; NOT PRESENT, SKIP ON
                     65    16   0043                JSB     (R5)                    ; CALL THRU THE VECTOR(S)
       55   00000000'GF    D0   0045        20$:    MOVL    G^EXE$GL_USRUNDWN, R5   ; GET SYSTEM-WIDE USER RUNDOWN VECTOR
                     02    13   004C                BEQL    30$                     ; NOT PRESENT, SKIP ON
                     65    16   004E                JSB     (R5)                    ; CALL THRU THE VECTOR(S)
                                0050        30$:                                    ;
                                0050        ;---------------------------------------------------------------
                                0050        ;
                                0050        ;       WRITE IMAGE ACCOUNTING RECORD
                                0050        ;
                                0050        ;---------------------------------------------------------------
   05 00000000'EF    01    E1   0050                BBC     #ACM$V_IMAGE,EXE$GL_ACMFLAGS,40$; IMAGE ACCOUNTING ENABLED ?
                     55    D4   0058                CLRL    R5                                ; NOT A SPECIAL KERNEL AST
                  FFA3'    30   005A                BSBW    EXE$IMGDELMSG                     ; WRITE IMAGE DELETION MESSAGE
                                005D        40$:
            00000000'9F    7C   005D                CLRQ    @#CTL$GQ_ISTART                   ; IMAGE ACCOUNTING INACTIVE
                                0063        ;---------------------------------------------------------------
                                0063        ;
                                0063        ;       CLOCK THE IMAGE COUNTER
                                0063        ;
                                0063        ;---------------------------------------------------------------
                                0063
       55   00000000'9F    D0   0063                MOVL    @#CTL$GL_PHD,R5         ; GET ADDRESS OF PROCESS HEADER
                OOF4 C5    D6   006A                INCL    PHD$L_IMGCNT(R5)        ; PROVIDE A CHECKING MECHANISM TO
                           .    006E                                                ; PREVENT DELIVERY OF AST'S TO THE
                                006E                                                ; WRONG OR SUBSEQUENT IMAGES
                                006E        ;---------------------------------------------------------------
                                006E        ;
                                006E        ;       RESET THE DISPATCH VECTORS TO THEIR INITIAL VALUES. (NO VECTORS USED)
                                006E        ;
                                006E        ;---------------------------------------------------------------
                                006E
            0000028D'EF    16   006E                JSB     EXE$RESETVEC            ; RESET DISPATCH VECTORS
00000000'GF 00000000'GF    D0   0074                MOVL    G^MMG$GL_RMSBASE,G^CTL$GL_RMSBASE ;RESET TO DEFAULT RMS
                                007F
                                007F        ;---------------------------------------------------------------
                                007F        ;
                                007F        ;       PAGE FAULT MONITORING RUNDOWN
                                007F        ;
                                007F        ;---------------------------------------------------------------
                                007F
                                007F                $SETPFM_S PFMFLG=#0             ; Terminate page fault monitoring
                                008C
                                008C        ;---------------------------------------------------------------
                                008C        ;
                                008C        ;       SECURITY AUDITING PROCESS RUNDOWN
                                008C        ;
                                008C        ;---------------------------------------------------------------
                                008C
                57    D5  008C   268            TSTL    R7                              ; IS THIS PROCESS RUNDOWN?
                15    12  008E   269            BNEQU   DASSIGN1                        ; BR IF NOT
     55   000004B6'9F     D0  0090   270            MOVL    @#NSA$T_IDT+NSA$L_IDT_AUDIT_CHAN,R5 ; IS JOURNAL ASSIGNED?
                0C    13  0097   271            BEQLU   DASSIGN1                        ; BR IF NOT
                          0099   272            $DEASJNL_S R5                           ; DEASSIGN JOURNAL
                          00A5   273
                          00A5   274    ;-----------------------------------------------------------
                          00A5   275    ;
                          00A5   276    ;       DEASSIGN CHANNELS WITHOUT OPEN FILES
                          00A5   277    ;
                          00A5   278    ;-----------------------------------------------------------
                          00A5   279    DASSIGN1:                                       ; FIRST DEASSIGN PASS
     55   00000000'9F     3C  00A5   280            MOVZWL  @#CTL$GW_CHINDX,R5              ; GET MAXIMUM INDEX
                26    13  00AC   281            BEQL    21$                             ; NO CHANNELS ASSIGNED, LEAVE
           55   55    CE  00AE   282            MNEGL   R5,R5                           ; CONVERT TO NEGATIVE OFFSET
                08    C1  00B1   283            ADDL3   #CCB$B_STS,-
     53   00000000'9F         00B3   284                    @#CTL$GL_CCBBASE,R3             ; COMPUTE ADDRESS OF HIGHEST CHANNEL
                          00B9   285            ASSUME  CCB$B_STS+1 EQ CCB$B_AMOD ;
       57   01 A345   91  00B9   286    10$:    CMPB    1(R3)[R5],R7                    ; IS THIS CHANNEL DEASSIGNABLE?
                05    14  00BE   287            BGTR    15$                             ; YES, ASSIGNED BY HIGHER ACMODE
                          00C0   288
                          00C0   289    ;
                          00C0   290    ; NOTE THAT THE PRIVILEGE CHECK TO DETERMINE WHETHER THE CHANNEL SHOULD
                          00C0   291    ; BE DEASSIGNED BY THIS MODE MUST BE A SIGNED COMPARISON.
                          00C0   292    ; THE F11BXQP 'RESERVES' A CHANNEL FOR ITSELF (NEVER TO BE USED BY ANYTHING
                          00C0   293    ; ELSE, OR DEASSIGNED) BY STORING A MINUS 1 (-1) IN THE CCB$B_AMOD FIELD.
                          00C0   294    ; THIS CAUSES A SIGNED TEST AGAINST EVEN KERNEL MODE TO FAIL AS IF THE
                          00C0   295    ; CHANNEL WERE ASSIGNED BY A HIGHER MODE.
                          00C0   296    ; THE CHANNEL NEVER NEEDS TO BE DEASSIGNED BECAUSE THERE IS NO SPECIFIC
                          00C0   297    ; DEVICE ASSOCIATED WITH IT - IT IS FABRICATED BY THE XQP INITIALIZATION
                          00C0   298    ; CODE WHEN THE PROCESS IS CREATED.
                          00C0   299    ;
                          00C0   300
      0A 6345   01    E1  00C0   301            BBC     #CCB$V_IMGTMP,(R3)[R5],20$ ; BRANCH IF NOT IMAGE TEMPORARY
           7E   55    CE  00C5   302    15$:    MNEGL   R5,-(SP)                        ; CONVERT TO REAL CHANNEL NUMBER
  00000000'GF   01    FB  00C8   303            CALLS   #1,G^SYS$DASSGN                 ; DE-ASSIGN CHANNEL
           55   10    C0  00CF   304    20$:    ADDL2   #CCB$C_LENGTH,R5                ; POINT AT NEXT CHANNEL
                E5    19  00D2   305            BLSS    10$                             ; NEXT CHANNEL
                          00D4   306
                          00D4   307    ;-----------------------------------------------------------
                          00D4   308    ;
                          00D4   309    ;       Deallocate the RDI table and zero CTL$GL_RDIPTR
                          00D4   310    ;
                          00D4   311    ;-----------------------------------------------------------
                          00D4   312
     50   00000000'9F     D0  00D4   313    21$:    MOVL    @#CTL$GL_RDIPTR,R0              ; GET RDI POINTER
                0F    13  00DB   314            BEQLU   22$                             ; BR IF NOT ALLOCATED
           51   60    D0  00DD   315            MOVL    RDI$L_SIZE(R0),R1               ; GET SIZE OF ALLOCATED BLOCK
     00000000'9F      D4  00E0   316            CLRL    @#CTL$GL_RDIPTR                 ; ZERO RDI POINTER
     00000000'GF      16  00E6   317            JSB     G^EXE$DEAP1                     ; DEALLOCATE RDI BLOCK
                          00EC   318    22$:
                          00EC   319
                          00EC   320    ;-----------------------------------------------------------
                          00EC   321    ;
                          00EC   322    ;       RELEASE MEMORY
                          00EC   323    ;
                          00EC   324    ;-----------------------------------------------------------
```

```
                              00EC    325
             FF11'   30       00EC    326             BSBW    MMG$IMGRESET                ; RELEASE IMAGE PAGES
                              00EF    327
                              00EF    328   ;-----------------------------------------------------------------
                              00EF    329   ;
                              00EF    330   ;          DEASSIGN CHANNELS - ALL
                              00EF    331   ;
                              00EF    332   ;-----------------------------------------------------------------
                              00EF    333
  55    00000000'9F    3C     00EF    334             MOVZWL  @#CTL$GW_CHINDX,R5          ; GET MAXIMUM INDEX
                30     13     00F6    335             BEQL    40$                         ; NO CHANNELS ASSIGNED, LEAVE
          55    55     CE     00F8    336             MNEGL   R5,R5                       ; CONVERT TO NEGATIVE OFFSET
                09     C1     00FB    337             ADDL3   #CCB$B_AMOD,-
  53    00000000'9F           00FD    338                     @#CTL$GL_CCBBASE,R3         ; COMPUTE ADDRESS OF HIGHEST CHANNEL
           57   6345   91     0103    339   25$:      CMPB    (R3)[R5],R7                 ; IS THIS CHANNEL DEASSIGNABLE?
                1A     15     0107    340             BLEQ    30$                         ; NO, ASSIGNED BY LOWER ACMODE
                              0109    341
                              0109    342   ;
                              0109    343   ; NOTE THAT THE PRIVILEGE CHECK TO DETERMINE WHETHER THE CHANNEL SHOULD
                              0109    344   ; BE DEASSIGNED BY THIS MODE MUST BE A SIGNED COMPARISON.
                              0109    345   ; THE F11BXQP 'RESERVES' A CHANNEL FOR ITSELF (NEVER TO BE USED BY ANYTHING
                              0109    346   ; ELSE, OR DEASSIGNED) BY STORING A MINUS 1 (-1) IN THE CCB$B_AMOD FIELD.
                              0109    347   ; THIS CAUSES A SIGNED TEST AGAINST EVEN KERNEL MODE TO FAIL AS IF THE
                              0109    348   ; CHANNEL WERE ASSIGNED BY A HIGHER MODE.
                              0109    349   ; THE CHANNEL NEVER NEEDS TO BE DEASSIGNED BECAUSE THERE IS NO SPECIFIC
                              0109    350   ; DEVICE ASSOCIATED WITH IT - IT IS FABRICATED BY THE XQP INITIALIZATION
                              0109    351   ; CODE WHEN THE PROCESS IS CREATED.
                              0109    352   ;
                              0109    353
           7E   55     CE     0109    354             MNEGL   R5,-(SP)                    ; CONVERT TO REAL CHANNEL NUMBER
   00000000'GF  01     FB     010C    355             CALLS   #1,G^SYS$DASSGN             ; DE-ASSIGN CHANNEL
           0D   50     E8     0113    356             BLBS    R0,30$                      ; BRANCH ON SUCCESS
   0000026C 8F  50     D1     0116    357             CMPL    R0,#SS$_IVCHNLSEC           ; CHECK FOR SECTION ON CHANNEL
                04     13     011D    358             BEQL    30$                         ; WHICH IS OK
                              011F    359             BUG_CHECK FILCNTNONZ,FATAL          ; IT REALLY SHOULD HAVE WORKED
                              0123    360
          55    10     C0     0123    361   30$:      ADDL    #CCB$C_LENGTH,R5            ; POINT AT NEXT CHANNEL
                DB     19     0126    362             BLSS    25$                         ; NEXT CHANNEL
                              0128    363
                              0128    364   ;-----------------------------------------------------------------
                              0128    365   ;
                              0128    366   ;          DEALLOCATE ALLOCATED DEVICES
                              0128    367   ;
                              0128    368   ;-----------------------------------------------------------------
                              0128    369
                              0128    370   40$:      $DALLOC_S       ACMODE=R7           ; DE-ALLOCATE AT ACCESS MODE
                              0133    371
                              0133    372   ;-----------------------------------------------------------------
                              0133    373   ;
                              0133    374   ;          CANCEL TIMER AND SCHEDULED WAKEUP REQUESTS
                              0133    375   ;
                              0133    376   ;-----------------------------------------------------------------
                              0133    377
                              0133    378             $CANTIM_S       ACMODE=R7           ; CANCEL TIMER REQUESTS
                              013E    379             $CANWAK_S                           ; CANCEL WAKE UP REQUESTS
                              0149    380
                              0149    381   ;-----------------------------------------------------------------
```

```
                                0149    ;
                                0149    ;       DEQUEUE REMAINING LOCKS
                                0149    ;
                                0149    ;---------------------------------------------------------------
                                0149
                                0149            $DEQ_S          ACMODE=R7,-             ; DEQUEUE ALL LOCKS AT SPECIFIED
                                0149                            FLAGS=#LCK$M_DEQALL!LCK$M_INVVALBLK
                                0158                                                    ; ACCESS MODE AND INVALIDATE
                                0158                                                    ; VALUE BLOCKS
                                0158
                                0158    ;---------------------------------------------------------------
                                0158    ;
                                0158    ;       DISASSOCIATE COMMON EVENT BLOCKS
                                0158    ;
                                0158    ;---------------------------------------------------------------
                                0158
                                0158            $DACEFC_S       EFN=#64                 ; DISASSOCIATE GROUP #2
                                0165            $DACEFC_S       EFN=#96                 ; DISASSOCIATE GROUP #3
                                0172
                                0172    ;---------------------------------------------------------------
                                0172    ;
                                0172    ;       CLEAR THE IPAST VECTOR AND ADDRESS
                                0172    ;
                                0172    ;---------------------------------------------------------------
                                0172
          51     0110 C4     D0 0172            MOVL    PCB$L_IPAST(R4),R1      ; Pick up the vector
                       24    13 0177            BEQL    70$                     ; There are none
                 52    07    D0 0179            MOVL    #7,R2                   ; Set up for 8 entries
          50     52    02    78 017C    50$:    ASHL    #2,R2,R0                ; Each entry is 4 bits
    53    51     04    50    EF 0180            EXTZV   R0,#4,R1,R3             ; Pick up the entry (mode+1)
                       13    13 0185            BEQL    60$                     ; None there
                 57    53    D1 0187            CMPL    R3,R7                   ; At a more priv mode than ACMODE?
                       0E    15 018A            BLEQ    60$                     ; Yes, do not clear
0110 C4   04     50    00    F0 018C            INSV    #0,R0,#4,PCB$L_IPAST(R4); No, clear it out
      00000000'9F42          D4 0193            CLRL    @#CTL$AL_IPASTVEC[R2]   ;  and its associated address
              DF 52          F4 019A    60$:    SOBGEQ  R2,50$                  ; Loop until all done
                                019D    70$:
                                019D
                                019D    ;---------------------------------------------------------------
                                019D    ;
                                019D    ;       PURGE PENDING UNDELIVERED AST CONTROL BLOCKS
                                019D    ;
                                019D    ;---------------------------------------------------------------
                                019D
                                019D    ASTRUNDWN:                              ;
          56     10 A4       DE 019D            MOVAL   PCB$L_ASTQFL(R4),R6     ; COMPUTE ADDRESS OF HEADER
                                01A1    LOCK_BEGIN:
                                01A1            SETIPL  W^LOCK_IPL              ; BLOCK AST ARRIVAL & LOCK DOWN CODE
    60 A4     00000004'GF    D1 01A6            CMPL    G^EXE$GQ_ERLMBX+4,PCB$L_PID(R4) ; CHECK FOR ERROR LOG MAILBOX
                       06    12 01AE            BNEQ    10$                     ; BR IF NO
          00000000'GF        7C 01B0            CLRQ    G^EXE$GQ_ERLMBX         ; YES, DEASSIGN IT
          50     14 A4       D0 01B6    10$:    MOVL    PCB$L_ASTQBL(R4),R0     ; GET TAIL OF AST QUEUE
                 56    50    D1 01BA            CMPL    R0,R6                   ; CHECK FOR EMPTY QUEUE
                       39    13 01BD            BEQL    20$                     ; YES, DONE WITH ASTS
    57    0B A0     02    00 ED 01BF            CMPZV   #ACB$V_MODE,#ACB$S_MODE,ACB$B_RMOD(R0),R7 ; CHECK ACCESS MODE OF ACB
                       31    19 01C5            BLSS    20$                     ; REACHED THE END OF DELETABLE ASTS
                 50    60    0F 01C7            REMQUE  (R0),R0                 ; REMOVE IT FROM AST QUEUE
                 2C    1D  01CA   439          BVS     20$                         ; EXIT IF QUEUE EMPTY
        03 0B A0 06    E1  01CC   440          BBC     #ACB$V_QUOTA,ACB$B_RMOD(R0),15$ ; SKIP IF NO QUOTA ACCOUNTING
              38 A4    B6  01D1   441          INCW    PCB$W_ASTCNT(R4)            ; ALLOW ANOTHER AST
           0B A0 30    93  01D4   442 15$:     BITB    #<ACB$M_NODELETE!ACB$M_PKAST>,ACB$B_RMOD(R0)  ; SPECIAL ACTIONS?
                 08    12  01D8   443          BNEQ    18$                         ; YES
        00000000'GF    16  01DA   444          JSB     G^EXE$DEANONPAGED           ; NO, DEALLOCATE IT
                 D4    11  01E0   445          BRB     10$                         ; CONTINUE
                           01E2   446
        CF 0B A0 04    E1  01E2   447 18$:     BBC     #ACB$V_PKAST,ACB$B_RMOD(R0),10$ ; BR IF NO PIGGYBACK AST
              55 50    D0  01E7   448          MOVL    R0,R5                       ; PUT ACB ADDRESS IN R5
                           01EA   449 ;
                           01EA   450 ; CALL PIGGY-BACK SPECIAL KERNEL AST ROUTINE.
                           01EA   451 ;        R5 - ACB ADDRESS (MUST BE PRESERVED)
                           01EA   452 ;        IPL = IPL$_ASTDEL (MUST NOT BE LOWERED)
                           01EA   453 ;
                           01EA   454          SETIPL  #IPL$_ASTDEL                ; LOWER IPL TO IPL$_ASTDEL
              5E 04    C2  01ED   455          SUBL    #4,SP                       ; EXTRA LONGWORD FOR INTER-PROCESS ASTS
              18 B5    16  01F0   456          JSB     @ACB$L_KAST(R5)             ; CALL KAST ROUTINE
              5E 04    C0  01F3   457          ADDL    #4,SP                       ; REMOVE EXTRA LONGWORD
                 A9    11  01F6   458          BRB     LOCK_BEGIN                  ; NO DELETE FOR PKAST
                           01F8   459
                           01F8   460 ;------------------------------------------------------------------
                           01F8   461 ;
                           01F8   462 ;        RESET AST ACTIVE AND AST ENABLE BITS
                           01F8   463 ;
                           01F8   464 ;------------------------------------------------------------------
                           01F8   465
              55 57    D0  01F8   466 20$:     MOVL    R7,R5                       ; INITIALIZE MODE INDEX
                 13    13  01FB   467          BEQL    40$                         ; SKIP TERMINATION HANDLER FOR KERNEL
                           01FD   468 30$:
              55 02    D1  01FD   469          CMPL    #2,R5                       ; SUPERVISOR OR USER MODE?
                 07    18  0200   470          BGEQ    35$                         ; BRANCH IF NOT
    FFFFFFF8'9F45      D4  0202   471          CLRL    @#CTL$GL_CMSUPR-8[R5]       ; CLEAR CHANGE MODE HANDLER
    FFFFFFFC'9F45      D4  0209   472 35$:     CLRL    @#CTL$GL_THEXEC-4[R5]       ; CLEAR TERMINATION HANDLER
    00000000'9F45      7C  0210   473 40$:     CLRQ    @#CTL$AQ_EXCVEC[R5]         ; CLEAR EXCEPTION HANDLER ENTRIES
    00000000'9F45      D4  0217   474          CLRL    @#CTL$AL_FINALEXC[R5]       ; CLEAR LAST CHANCE EXCEPTION HANDLER
        00 0D A4 55    E2  021E   475          BBSS    R5,PCB$B_ASTEN(R4),50$      ; SET ENABLE BIT
        00 0C A4 55    E5  0223   476 50$:     BBCC    R5,PCB$B_ASTACT(R4),60$     ; CLEAR AST ACTIVE
         50   55 06    C1  0228   477 60$:     ADDL3   #PCB$V_SSFEXC,R5,R0         ; COMPUTE SSFAIL ENABLE BIT NUMBER
        00 24 A4 50    E5  022C   478          BBCC    R0,PCB$L_STS(R4),70$        ; AND CLEAR IT
           C8 55 04    F2  0231   479 70$:     AOBLSS  #4,R5,30$                   ; DO ALL REQUESTED MODES
        00000000'9F    D4  0235   480          CLRL    @#CTL$GL_CMHANDLR           ; CLEAR COMPATIBILITY MODE HANDLER
        00000000'GF    16  023B   481          JSB     G^SCH$NEWLVL                ; COMPUTE NEW ASTLVL
        00001004 8F    CA  0241   482          BICL    #<1@PCB$V_FORCPEN>!<1@PCB$V_WAKEPEN>,- ; CLEAR FORCE EXIT AND
              24 A4        0247   483                  PCB$L_STS(R4)               ; WAKE PENDING
                           0249   484          SETIPL  #0                          ; ENABLE AGAIN
                           024C   485 ;------------------------------------------------------------------
                           024C   486 ;
                           024C   487 ;        DELETE IMAGE LOGICAL NAMES
                           024C   488 ;
                           024C   489 ;------------------------------------------------------------------
                           024C   490
                           024C   491 LOGNRUNDWN:                                  ; LOGICAL NAME RUNDOWN
              52 57    D0  024C   492          MOVL    R7,R2                       ; ACCESS MODE ARG TO RIGHT REGISTER
  53    00000000'9F    D0  024F   493          MOVL    @#CTL$GL_LNMHASH,R3         ; ADDRESS OF PROCESS HASH TABLE
                 06    13  0256   494          BEQL    5$                          ; NEVER ALLOCATED SPACE
        00000000'GF    16  0258   495          JSB     G^LNM$DELETE_HASH           ; DELETE ALL NAMES
```

```
                                  025E  5$:
                                  025E  ;--------------------------------------------------------------
                                  025E  ;
                                  025E  ;       Reset P0 extention to Process Allocation Region
                                  025E  ;
                                  025E  ;--------------------------------------------------------------
                                  025E
           00000000'GF        7C  025E         CLRQ    G^CTL$GQ_P0ALLOC              ; RESET LIST HEAD TO NULL
        50 00000000'GF        3C  0264         MOVZWL  G^SGN$GW_CTLIMGLIM,R0         ; GET IMAGE LIMIT ON P0 REQUESTS
00000000'GF        50     09  78  026B         ASHL    #9,R0,G^CTL$GL_PRCALLCNT;         AND RESET CELL CONTENTS.
                                  0273
                                  0273  ;--------------------------------------------------------------
                                  0273  ;
                                  0273  ;       Restore previous setting of resource wait mode
                                  0273  ;
                                  0273  ;--------------------------------------------------------------
                                  0273
                          50  D4  0273         CLRL    R0                            ; ASSUME ON
                  09      8E  D1  0275         CMPL    (SP)+,#SS$_WASSET             ; CHECK IF IT WAS OFF
                          02  12  0278         BNEQ    10$                           ; BRANCH IF ON
                          50  D6  027A         INCL    R0                            ; SET TO TURN OFF
                                  027C  10$:   $SETRWM_S WATFLG=R0
                                  0285
                  50      01  3C  0285         MOVZWL  #SS$_NORMAL,R0                ; SET NORMAL COMPLETION STATUS
                              04  0288         RET                                   ; RETURN
                                  0289
                                  0289  LOCK_IPL:                                    ; END OF LOCKED DOWN CODE
                        00000008  0289         .LONG   IPL$_SYNCH
                                  028D  LOCK_END:
                                  028D         ASSUME  LOCK_END-LOCK_BEGIN LE 512         ; MUST BE ON ADJOINING PAGES
```

```
SYS
Sym

ACB
AST
BUI
EFN
EXE
EXE
EXE
EXE
EXE
EXE
EXE
IPL
IPL
IPL
JIB
LOG
PCB
PCB
PCB
PCB
PCB
PCB
PID
PR$
PSL
PSL
REQ
RSN
SCH
SCH
SCH
SCH
SCH
SS$
SS$
SS$
SS$
TMD
TQE
TQE
TQE
TQE
TQE
TQE
TQE
TQE
TQE
TQE
TQE
TQE
WKD
WKR
```

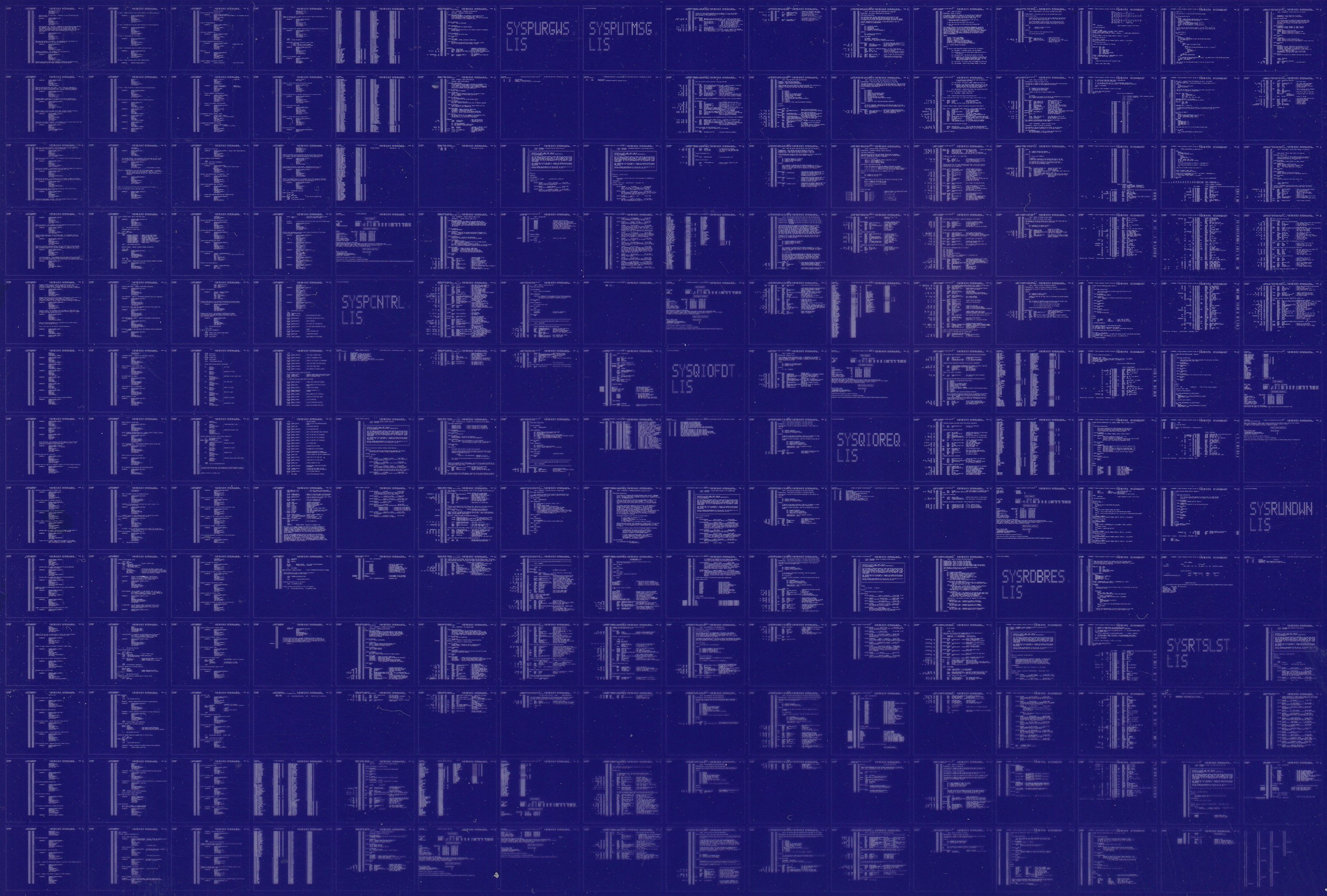
```
                               028D             .SBTTL  EXE$RESETVEC - REST PROVILEGED LIBRARY VECTORS
                               028D
                               028D     ;++
                               028D     ; FUNCTIONAL DESCRIPTION:
                               028D     ;       THIS ROUTINE RESETS THE KERNEL AND EXEC CHANGE MODE DISPATCH VECTORS
                               028D     ;       ALONG WITH THE RUNDOWN VECTORS.
                               028D     ;
                               028D     ; CALLING SEQUENCE:
                               028D     ;       JSB EXE$RESETVEC
                               028D     ;
                               028D     ;       R0, R1, R2, R3, AND R6 ARE DESTROYED
                               028D     ;
                               028D     ;
                               028D     ;--
                               028D
                               028D     EXE$RESETVEC::
  56   00000000'9F     9E      028D             MOVAB   @#CTL$A_DISPVEC,R6      ;GET ADR OF 1ST VECTOR PAGE
                               0294     ; VERIFY THAT EACH SET OF VECTORS USES ONE HALF PAGE
                               0294
                               0294             ASSUME  <NXTEVEC-NXTKVEC> EQ 256
                               0294             ASSUME  <NXTRVEC-NXTEVEC> EQ 256
                               0294             ASSUME  <NXTMVEC-NXTRVEC> EQ 256
                               0294
          52    04     9A      0294             MOVZBL  #4,R2                   ; THERE ARE FOUR VECTORS TO RESET
  51   00000000'9F     3E      0297             MOVAW   @#IAC$AW_VECRESET,R1    ; R1 LOCATES RESET CONTEXT
  53   00000000'9F     3E      029E             MOVAW   @#IAC$AW_VECSET,R3      ; R3 LOCATES THE STARTING POINT
                               02A5
          50    61     3C      02A5     50$:    MOVZWL  (R1),R0                 ; GET OFFSET INTO VECTOR
          83    81     B0      02A8             MOVW    (R1)+,(R3)+             ; RESET STARTING POINT
          66    50     D0      02AB             MOVL    R0,(R6)                 ; STORE OFFSET INTO FIRST LONGWORD
          50    56     C0      02AE             ADDL2   R6,R0                   ; LOCATE ADDRESS TO BE RESET
          60    05     9A      02B1             MOVZBL  #OP$_RSB,(R0)           ; STORE "RSB" AT APPROPRIATE OFFSET
     56   0100  C6     DE      02B4             MOVAL   256(R6),R6              ; ADVANCE R6 TO NEXT VECTOR
             E9 52     F5      02B9             SOBGTR  R2,50$                  ;  AND LOOP IF NOT YET DONE
                05             02BC             RSB
                               02BD
                               02BD             .END
```

```
$$T1                = 00000000             MMG$GL_RMSBASE            ********   X   02
ACB$B_RMOD          = 0000000B             MMG$IMGRESET              ********   X   02
ACB$L_KAST          = 00000018             NSA$L_IDT_AUDIT_CHAN    = 000004B6
ACB$M_NODELETE      = 00000020             NSA$T_IDT                 ********   X   02
ACB$M_PKAST         = 00000010             NXTEVEC                 = 00000100
ACB$S_MODE          = 00000002             NXTKVEC                 = 00000000
ACB$V_MODE          = 00000000             NXTMVEC                 = 00000300
ACB$V_PKAST         = 00000004             NXTRVEC                 = 00000200
ACB$V_QUOTA         = 00000006             OP$_RSB                 = 00000005
ACM$V_IMAGE         = 00000001             PCB$B_ASTACT            = 0000000C
ACMODE              = 00000004             PCB$B_ASTEN             = 0000000D
ASTRUNDWN             0000019D R     02    PCB$L_ASTQBL            = 00000014
BUG$_FILCNTNONZ       ********   X   02    PCB$L_ASTQFL            = 00000010
CCB$B_AMOD          = 00000009             PCB$L_IPAST             = 00000110
CCB$B_STS           = 00000008             PCB$L_PID               = 00000060
CCB$C_LENGTH        = 00000010             PCB$L_STS               = 00000024
CCB$V_IMGTMP        = 00000001             PCB$V_FORCPEN           = 00000002
CJF$DEASJNL           ********  GX   02    PCB$V_PWRAST            = 00000016
CTL$AL_FINALEXC       ********   X   02    PCB$V_SSFEXC            = 00000006
CTL$AL_IPASTVEC       ********   X   02    PCB$V_WAKEPEN           = 0000000C
CTL$AQ_EXCVEC         ********   X   02    PCB$W_ASTCNT            = 00000038
CTL$A_DISPVEC         ********   X   02    PHD$L_IMGCNT            = 000000F4
CTL$GL_CCBBASE        ********   X   02    PR$_IPL                 = 00000012
CTL$GL_CMHANDLR       ********   X   02    PROC_TABLE                00000000 R     02
CTL$GL_CMSUPR         ********   X   02    RDI$C_SIZE              = 00000000
CTL$GL_LNMHASH        ********   X   02    SCH$NEWLVL                ********   X   02
CTL$GL_PHD            ********   X   02    SGN$GW_CTLIMGLIM          ********   X   02
CTL$GL_POWERAST       ********   X   02    SS$_IVCHNLSEC           = 0000026C
CTL$GL_PRCALLCNT      ********   X   02    SS$_NORMAL              = 00000001
CTL$GL_RDIPTR         ********   X   02    SS$_WASSET              = 00000009
CTL$GL_RMSBASE        ********   X   02    SYS$CANTIM                ********  GX   02
CTL$GL_THEXEC         ********   X   02    SYS$CANWAK                ********  GX   02
CTL$GL_USRUNDWN       ********   X   02    SYS$DACEFC                ********  GX   02
CTL$GQ_ISTART         ********   X   02    SYS$DALLOC                ********  GX   02
CTL$GQ_P0ALLOC        ********   X   02    SYS$DASSGN                ********   X   02
CTL$GW_CHINDX         ********   X   02    SYS$DEQ                   ********  GX   02
DASSIGN1              000000A5 R     02    SYS$SETPFM                ********  GX   02
EXE$DEANONPAGED       ********   X   02    SYS$SETRWM                ********  GX   02
EXE$DEAP1             ********   X   02
EXE$GL_ACMFLAGS       ********   X   02
EXE$GL_USRUNDWN       ********   X   02
EXE$GQ_ERLMBX         ********   X   02
EXE$IMGDELMSG         ********   X   02
EXE$MAXACMODE         ********   X   02
EXE$RESETVEC          0000028D RG    02
EXE$RUNDWN            00000013 RG    02
IAC$AW_VECRESET       ********   X   02
IAC$AW_VECSET         ********   X   02
IPL$_ASTDEL         = 00000002
IPL$_SYNCH          = 00000008
LCK$M_DEQALL        = 00000001
LCK$M_INVVALBLK     = 00000004
LNM$DELETE_HASH       ********   X   02
LOCK_BEGIN            000001A1 R     02
LOCK_END              0000028D R     02
LOCK_IPL              00000289 R     02
LOGNRUNDWN            0000024C R     02
```

```
                                           +----------------+
                                           ! Psect synopsis !
                                           +----------------+

PSECT name                        Allocation           PSECT No.  Attributes
----------                        ----------           ---------  ----------
.  ABS  .                         00000000  (     0.)  00 (   0.)  NOPIC    USR    CON    ABS    LCL NOSHR NOEXE NORD   NOWRT NOVEC BYTE
$ABS$                             00000000  (     0.)  01 (   1.)  NOPIC    USR    CON    ABS    LCL NOSHR   EXE   RD     WRT NOVEC BYTE
YEXEPAGED                         000002BD  (   701.)  02 (   2.)  NOPIC    USR    CON    REL    LCL NOSHR   EXE   RD     WRT NOVEC BYTE

                                        +------------------------+
                                        ! Performance indicators !
                                        +------------------------+

Phase                          Page faults    CPU Time        Elapsed Time
-----                          -----------    --------        ------------
Initialization                          31    00:00:00.08     00:00:00.27
Command processing                     113    00:00:00.54     00:00:01.70
Pass 1                                 371    00:00:12.04     00:00:24.98
Symbol table sort                        0    00:00:01.97     00:00:04.28
Pass 2                                 117    00:00:02.54     00:00:05.51
Symbol table output                     11    00:00:00.09     00:00:00.24
Psect synopsis output                    2    00:00:00.02     00:00:00.08
Cross-reference output                   0    00:00:00.00     00:00:00.00
Assembler run totals                   647    00:00:17.29     00:00:37.07

The working set limit was 1650 pages.
71031 bytes (139 pages) of virtual memory were used to buffer the intermediate code.
There were 70 pages of symbol table space allocated to hold 1326 non-local and 28 local symbols.
563 source lines were read in Pass 1, producing 19 object records in Pass 2.
32 pages of virtual memory were used to define 31 macros.

                                       +--------------------------+
                                       ! Macro library statistics !
                                       +--------------------------+

Macro library name                               Macros defined
------------------                               --------------
_$255$DUA28:[SYS.OBJ]LIB.MLB;1                             10
_$255$DUA28:[SYSLIB]STARLET.MLB;2                          18
TOTALS (all libraries)                                     28

1460 GETS were required to define 28 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS$:SYSRUNDWN/OBJ=OBJ$:SYSRUNDWN MSRC$:SYSRUNDWN/UPDATE=(ENH$:SYSRUNDWN)+EXECML$/LIB
```

0387 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



0388 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY